

AD-A040 546

TEXAS UNIV AT AUSTIN CENTER FOR CYBERNETIC STUDIES

F/G 15/5

THE GENERALIZED ALTERNATING PATH ALGORITHM FOR TRANSPORTATION P--ETC(U)

MAR 77 R S BARR, F GLOVER, D KLINGMAN

N00014-76-C-0383

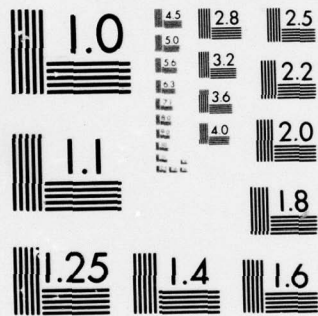
NL

UNCLASSIFIED

CCS-282

| OF |
AD
A040546





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

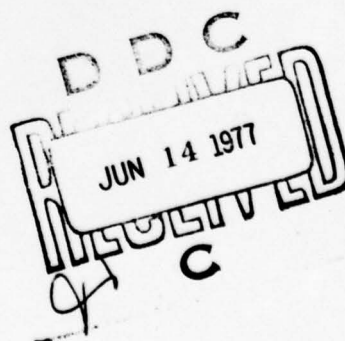
ADA040546

12
NW



CENTER FOR CYBERNETIC STUDIES

The University of Texas
Austin, Texas 78712



DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited



9 14
Research Report, CCS-282

6 THE GENERALIZED ALTERNATING PATH
ALGORITHM FOR TRANSPORTATION PROBLEMS,

10 by

R. S. Barr*
F. Glover**
D. Klingman***

15 N00014-76-C-0383,
N00014-75-C-0616

12 24 p.

11 March 1977

DDC
RECEIVED
JUN 14 1977
C

*Assistant Professor of Management Science, Department of Industrial Engineering and Operations Research, Southern Methodist University, Dallas, Texas 75275

**Professor of Management Science, College of Business Administration, University of Colorado, Boulder, Colorado 80302

***Professor of Operations Research and Computer Science, Department of General Business, BEB 600, University of Texas, Austin, Texas 78712

This research was partly supported by ONR Contract N00014-76-C-0383 with Decision Analysis and Research Institute and by Project NR047-021, ONR Contracts N00014-75-C-0616 and N00014-75-C-0569 with the Center for Cybernetic Studies, The University of Texas. Reproduction in whole or in part is permitted for any purpose of the United States Government.

CENTER FOR CYBERNETIC STUDIES

A. Charnes, Director
Business-Economics Building, 203E
The University of Texas
Austin, Texas 78712
(512) 471-1821

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

406 197

ABSTRACT

A new primal extreme point algorithm for solving capacitated transportation problems is developed in this paper. This algorithm, called the generalized alternating path (GAP) algorithm, is a special purpose method specifically designed to take advantage of the bipartite structure and the often pervasive primal degeneracy of transportation problems.

NO. 10	
NO. 10	<input checked="" type="checkbox"/>
NO. 10	<input type="checkbox"/>
NO. 10	<input type="checkbox"/>
DISPATCH/AVAILABILITY CODES	
NO. 10	NO. 10
NO. 10	NO. 10

1. INTRODUCTION

The purpose of this paper is to present a primal extreme point algorithm for solving transportation problems which both circumvents and exploits degeneracy. The algorithm, called the generalized alternating path (GAP) algorithm, is an extension of the algorithms presented in [5, 6] and a specialization of the algorithm presented in [9]. We have undertaken in this paper to fill the vacant space between more general and more specialized solution procedures because of the importance of transportation problems as a problem class, and also because of unique structural features of these problems that require correspondingly unique adaptations to be handled efficiently. In particular, our development focuses on relationships that assume a special form for the transportation problem, and on their implications for implementation.

The generalized alternating path algorithm is based on the characterization of a special type of basis, called the GAP basis. The GAP bases comprise a subset of the bases that are capable of leading to an optimal solution if one exists. We show by a particularly simple proof that it is sufficient to examine only the bases that lie in this subset at each iteration of the algorithm.

From a practical standpoint, we further demonstrate how the graphical structure of these bases can be used to streamline the computer implementation of the procedure. Thus, the GAP algorithm has readily identifiable computational advantages over other specializations of the primal simplex algorithm to transportation problems [11, 12]. These procedural innovations are also particularly relevant to recent efforts in the literature [14, 16] to identify the merits of alternative approaches to exploiting bipartite network structures.

2. BACKGROUND MATERIAL

A capacitated $m \times n$ transportation problem may be defined as:

$$\text{Minimize } \sum_{(i,j) \in A} c_{ij} x_{ij}$$

subject to:

$$\sum_{j \in \{j: (i,j) \in A\}} x_{ij} = a_i, \quad i \in I = \{1, 2, \dots, m\}$$

$$\sum_{i \in \{i: (i,j) \in A\}} x_{ij} = b_j, \quad j \in J = \{1, 2, \dots, n\}$$

$$0 \leq x_{ij} \leq U_{ij}, \quad (i,j) \in A$$

where I is called the set of origin nodes, J is called the set of destination nodes, A is the set of admissible arcs, and c_{ij} is the cost of shipping a unit from origin node i to destination node j .

The dual of the capacitated transportation problem may be stated as:

$$\text{Maximize } \sum_{i \in I} a_i R_i + \sum_{j \in J} b_j K_j + \sum_{(i,j) \in A} U_{ij} W_{ij}$$

subject to:

$$R_i + K_j + W_{ij} \leq c_{ij}, \quad (i,j) \in A$$

$$W_{ij} \leq 0, \quad (i,j) \in A$$

where R_i and K_j are called the node potentials of the origin and destination nodes, respectively.

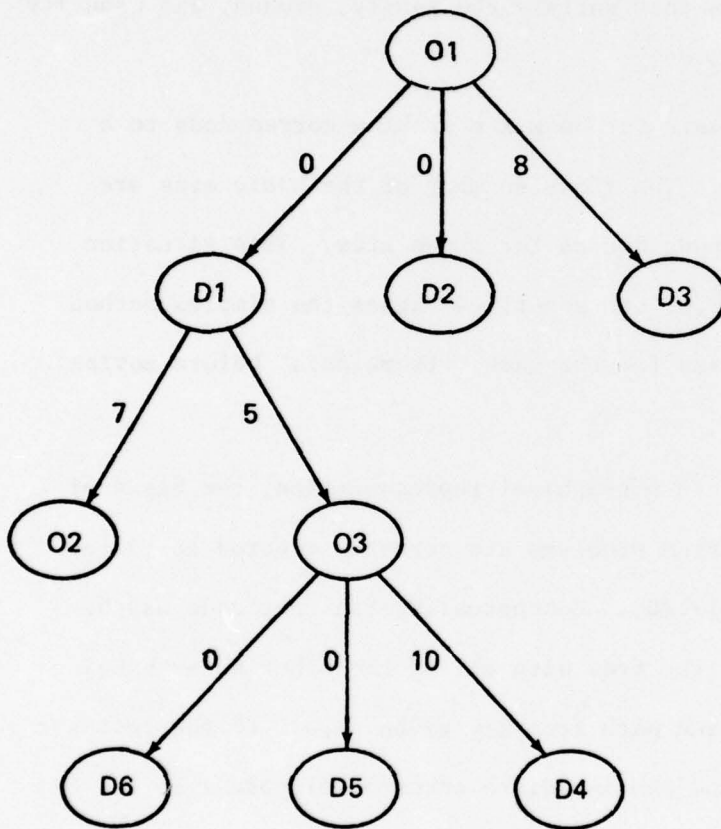
A familiarity with graphical interpretations of the transportation problem and the operations of the primal simplex method specialized to this framework is especially useful for understanding the results of this paper. We summarize these ideas in this section for completeness [1, 2, 4, 10, 12, 15, 19, 20]. At the same time, we will introduce terminology that will subsequently be used to characterize the GAP algorithm.

The transportation problem may be represented as a bipartite graph consisting of a set of origin nodes with supplies a_i and a set of destination nodes with demands b_j . Directed arcs from origin nodes to destination nodes accommodate the transmission of flow and incur a cost if flow exists. The objective is to determine a set of arc flows that satisfy the supply, demand, and capacity restrictions at minimum total cost.

A bounded variable simplex basis for an $m \times n$ problem corresponds to a spanning tree with $m + n - 1$ arcs. The flows on many of the basic arcs are frequently equal to zero or the upper bounds for these arcs. This situation provides a degenerate basic solution, and sometimes causes the simplex method to examine several alternative bases for the same extreme point before moving to an adjacent extreme point.

In solution procedures based on a graphical representation, the bases of the simplex method for transportation problems are normally treated as rooted trees [1,2,4,8,10,11,12,13,15,18,19,20]. Conceptually, the root node may be thought of as the highest node in the tree with all of the other nodes hanging below it. Those nodes in the unique path from any given node i to the root are called the ancestors of node i , and the immediate ancestor of node i is called its predecessor.

Figure 1 illustrates a rooted basis tree, the predecessors of the nodes, and the basic flow values, for a 3×6 transportation problem. Notationally, O_i denotes the i th origin node and D_j denotes the j th destination node. The number beside each link (arc) in the basis tree indicates the flow on this arc imparted by the basic solution. (Note that the nonbasic arcs at their upper bound, are not shown.) Predecessors of nodes are identified in the NODE/PREDECESSOR array. For example, as seen from this array, the predecessor of origin node 2 is destination node 1. The root of the tree is node O_1 and has no predecessor.



Node	Predecessor
O1	None
O2	D1
O3	D1
D1	O1
D2	O1
D3	O1
D4	O3
D5	O3
D6	O3

Figure 1 — Rooted Basis Tree

It is important to note that the direction of the links in Figure 1 correspond to the orientation induced by the predecessor ordering and do not necessarily correspond to the direction of the basis arcs in the transportation problem. However, the direction of the basic arcs are known from the bipartite property of the transportation problem; i.e., all problem arcs lead from origin nodes to destination nodes.

In subsequent sections the term O-D link and D-O link will be used to refer to links in a rooted basis tree that are directed from an origin node to a destination node and vice versa, according to the orientation imparted to the basic arcs by the predecessor indexing. For example, in Figure 1, 01-D1 is an O-D link while D1-02 is a D-O link. Additionally, basic arcs with a flow greater than 0 will be referred to as lower leeway links and basic arcs with a flow less than their upper bounds will be called upper leeway links. Basic arcs that are both lower and upper leeway links will be called double leeway links.

Basis Exchanges

The fundamental pivot, or basis exchange, step of the simplex method will now be briefly reviewed in the graphical setting. Assume that a feasible starting basis has been determined and is represented as a rooted tree. To evaluate the nonbasis arcs to determine whether any of them "price out" profitably, and therefore are candidates to enter the basis, it is necessary to determine values for the dual variables R_i , $i \in I$, and K_j , $j \in J$, which satisfy complementary slackness; i.e., which yield $R_i + K_j = c_{ij}$ for each basic arc.

There is a unique dual variable associated with each node in the basis tree. For this reason the dual variables--or their values--are often referred to as node potentials. Because of redundancy in the defining equations of the transportation problem (and in network problems generally), one node potential may be specified arbitrarily. The root node is customarily selected for this

purpose and assigned a potential of zero, whereupon the potentials of the other nodes are immediately determined in a cascading fashion by moving down the tree and identifying the value for each node from its predecessor using the equation $R_i + K_j = c_{ij}$. Highly efficient labeling procedures for traversing the tree to initialize and update these node potential values are described in [1,2,4,10,15,19,20].

A feasible basic solution is optimal if the updated cost coefficient $\pi_{ij} (= R_i + K_j - c_{ij})$ is nonpositive for all the nonbasic arcs with flow equal zero and nonnegative for all nonbasic arcs with flow equal to U_{ij} . If the solution is not optimal, then a nonbasic arc which violates the nonnegativity or nonpositivity requirement for π_{ij} is selected to enter the basis. If the flow on the selected arc is zero (U_{ij}), then the simplex method attempts to increase (decrease) this flow. The arc to leave the basis is determined by: (1) finding the unique path in the basis tree, called the basis equivalent path, which connects the two nodes of the entering arc, and (2) isolating a blocking arc in this path whose flow goes to zero or its upper bound ahead of (or at least as soon as) any others as a result of increasing or decreasing the flow on the entering arc.

In the basis equivalent path all arcs an even number of links away from the entering arc are called even arcs, and all arcs an odd number of links away are called odd arcs. (The incoming arc itself is considered the "0 arc," and hence is even). An increase (decrease) in the flow of the incoming arc causes a corresponding increase (decrease) in the flow of all even arcs and a corresponding decrease (increase) in the flow of all odd arcs. Thus, if an odd arc already has a 0 flow or an even arc already has a U_{ij} flow, then such an arc qualifies as a blocking arc and the incoming arc cannot be assigned a nonzero flow change.

To illustrate, assume that the starting basis is the one given in Figure 1 and the entering arc is (02,D2). The basis equivalent path for (02,D2) is D2-01-D1-02. Note that this path can be easily determined by tracing the predecessors of 02 and D2 to their point of intersection [10,13,20]. As flow is increased on the entering arc the flow on the arc (01,D2), which is an odd number of links away, must be decreased. However, its flow is already zero, and hence (01,D2) qualifies as a blocking arc. When arc (02,D2) is brought into the basis, arc (01,D2) must be dropped (since there are no other blocking arcs in this case). In addition, the pivot (or basis exchange) is degenerate since no flow increase occurs.

Once the entering and leaving arcs are known, the basis exchange is completed simply by updating the flow values on the basis equivalent path and determining new node potentials for the new basis tree. Only a subset of the node potentials change during a pivot and these can be updated rather than being determined from scratch.

To update the node potentials, assume that the nonbasic arc (p,q) is to enter into the basis and the basic arc (r,s) is to leave the basis. If arc (r,s) is deleted from the basis (before adding arc (p,q), two subtrees are formed, each containing one of the two nodes of the incoming arc (p,q). Let K denote the subtree which does not contain the root node of the full basis. The node potentials for the new basis may be obtained [10] by updating only those potentials of the nodes in K, as follows. If p is in K, subtract $\delta = R_p + K_q - c_{pq}$ from the potentials of each origin node in K and add δ to the potential of each destination node in K. Otherwise, q is in K and $-\delta$ is used in the above operations. (Note that $\delta > 0$ if arc (p,q) is nonbasic with zero flow and $\delta < 0$ if arc (p,q) is nonbasic with U_{pq} flow).

3. GENERALIZED ALTERNATING PATH BASIS DEFINITION AND CONSTRUCTION

The new generalized alternating path algorithm for transportation problems developed in this section is based upon the primal simplex method as described above, and extends the special alternating path algorithm for the assignment and semi-assignment problems [5,6]. The major mathematical distinction of the new method in contrast to the simplex method, is that it does not consider all feasible bases to be candidates for progressing to an optimal basis. That is, the simplex method allows a feasible spanning tree of any structure whatsoever to be included in the set of those that are eligible for consideration as "improving bases" along the path to optimality. However, it will be shown that if a transportation problem has an optimal solution then it also has an optimal solution with a unique basis tree structure, dubbed the generalized alternating path (GAP) structure. Furthermore, we will show that it is possible to restrict attention at each step to bases with this structure. In particular, the proposed algorithm is a procedure designed to exploit the properties of the GAP basis structure in a manner that substantially reduces the impact of degeneracy and the number of arithmetic operations required to solve the transportation problem.

Definition: A rooted basis tree for a capacitated transportation problem is a generalized alternating path (GAP) basis if

1. The root node is an origin node.
2. All O-D links are lower leeway links.
3. All D-O links are upper leeway links.

Definition: A generalized alternating path (GAP) is an elementary path such that all odd arcs are lower leeway links and all even arcs are upper leeway links, or vice versa. (No restriction is placed on what type of arc should appear first, or on the total number of arcs in the path.) Thus, in the GAP basis, every path from a given node to an ancestor is a generalized alternating path.

Lemma 1: Given an arbitrary feasible extreme point solution to a capacitated transportation problem, it is possible to identify a GAP basis, possibly including artificial arcs with 0 flows, that assigns the same flows to all admissible arcs.

Rather than prove the preceding lemma formally, we will establish its validity by sketching one of several possible ways to construct a GAP basis with the indicated property.

A GAP Basis Construction

1) To begin, exclude from consideration all arcs whose flows equal their upper or lower bounds. All remaining arcs are both lower leeway and upper leeway links. (These may be called double leeway links, as contrasted to the single leeway links that have been temporarily excluded from consideration.)

2) The subgraph induced by the remaining arcs is a forest (plus possibly some isolated nodes). Select any origin node as a root node for each tree in the forest, and establish the customary predecessor indexing [10, 13] for the tree. The result is a set of trees, each member of which has the GAP structure.

3) All isolated destination nodes can now be included in this forest as follows. There must be at least one saturated arc (i.e., whose flow equals its upper bound) that connects to any given isolated destination node (assuming without loss of generality that all $b_j \neq 0$). Add one such arc and orient it from its origin node to the destination node (i.e., making the origin node the predecessor). This preserves the GAP structure (assuming, again without loss of generality, that no $U_{ij} = 0$).

4) The trees of the forest and any isolated origin nodes may be strung together as follows. Establish the predecessor indexing for all trees. Select any two trees, or one tree and an isolated origin node. Then select an arc (possibly an

artificial arc) at its lower bound which connects the root of one of the trees (or the isolated origin node) to a destination node of the other tree. Orient this arc as a D-0 link, and then repeat the process until the entire graph is connected.

The result of the foregoing process is a GAP basis since each added arc in Step 4 has upper leeway, and further, the characterization of this arc as a D-0 link does not require that any previous arc change its orientation.

The GAP bases clearly constitute only a subset of those that correspond to a given degenerate extreme point solution--i.e., under degeneracy, many bases exist that cannot be transformed into a GAP basis by any orientation of the arcs or by swapping the designation of the origin nodes and the destination nodes. In fact, we may make the following observation.

Remark 1: A basis that satisfies any of the following properties cannot be transformed into a GAP basis: (1) there exists a node with two incident basic arcs with 0 flow and two incident basic arcs with saturating flow; (2) there exists any subpath of the basis of three successive arcs that all have 0 flows or all have saturating flow; (3) there exists any origin node and any destination node such that both have more than one incident 0-flow arc, or both have more than one incident saturated arc (this includes (2) as a special case).

The foregoing remark discloses that the GAP basis structure is indeed highly restrictive.

4. IMPORTANT PROPERTIES OF GAP BASES

We will show that for any choice of an incoming nonbasic arc in a basis exchange (pivot) step for a primal simplex method, there is a unique outgoing arc which can be selected to leave the basis that will maintain primal feasibility and also preserve the GAP basis structure. This unique arc must be

selected by a different rule in each of four different situations. To characterize these situations and their appropriate rules, we introduce the following terminology.

Arc (p,q) will denote the incoming arc. The unique intersection node on the predecessor paths from nodes Op and Dq to the root node will be denoted by Z , where Z may in some instances be equal to Op or Dq . The predecessor paths from Op to Z and Dq to Z will be denoted by $Z-Op$ and $Z-Dq$, respectively. Note that one of these two paths may consist of only the node Z itself.

It is convenient to augment each of the paths $Z-Op$ and $Z-Dq$ with arc (p,q) such that it is an $O-D$ link and $D-O$ link in the augmented paths $Z-Op$ and $Z-Dq$, respectively. The augmented $Z-Op$ and $Z-Dq$ paths will be denoted by $\overline{Z-Op}$ and $\overline{Z-Dq}$. The terms lower and higher links will be used to refer to arcs in the $\overline{Z-Dq}$ and $\overline{Z-Op}$ paths according to the natural orientation imparted by the predecessor indexing where the predecessor of any node X is considered to be above node X itself. Thus, the lowest link in each path is arc (p,q) and the highest link in each path are the arcs connected to node Z .

The rules for selecting the outgoing arc will now be itemized according to the relevant possibilities:

- I. The incoming arc (p,q) is a O -flow arc and $\pi_{pq} > 0$. (In this case the primal simplex method undertakes to increase the flow on arc (p,q) . If Δ is the amount of flow change, then all $D-O$ links on the $\overline{Z-Dq}$ path and all $O-D$ links on the $\overline{Z-Op}$ path increase their flow by Δ . All other arcs on the paths decrease their flow by Δ .) (Note that arc (p,q) is in fact a $D-O$ link on the $\overline{Z-Dq}$ path and an $O-D$ link on the $\overline{Z-Op}$ path--hence all arcs on a given path of the same type as (p,q) change their flow in the same way.)

A. If any arc on the $\overline{Z-Op}$ path reaches its upper or lower bound (i.e., blocks additional flow change) strictly before any arc on the $\overline{Z-Dq}$ path reaches its upper or lower bound, then the lowest arc on the $\overline{Z-Op}$ path that thus restricts the flow change is the outgoing arc.

B. If any arc on the $\overline{Z-Dq}$ path reaches its upper or lower bound, (i.e., blocks flow change) at the same time as or before any arc on the $\overline{Z-Op}$ path reaches its upper or lower bound, then the highest arc on the $\overline{Z-Dq}$ path that thus restricts the flow change is the outgoing arc. (Note that if arc (p,q) is one of the first arcs to hit its limit, then this automatically qualifies as case I.B.)

II. The incoming arc (p,q) is an arc with saturated flow and $\pi_{pq} < 0$. (The primal simplex method thus undertakes to decrease the flow on arc (p,q) . If the flow change is Δ (in absolute value) then all D-0 links on the $\overline{Z-Dq}$ path and all 0-D links on the $\overline{Z-Op}$ path are decreased by Δ . The remaining arcs on these paths increase their flow by Δ .)

A. If any arc on the $\overline{Z-Dq}$ path reaches its upper or lower bound (blocks flow change) strictly before any arc on the $\overline{Z-Op}$ path, then the lowest arc on the $\overline{Z-Dq}$ path that thus restricts the flow change is the outgoing arc.

B. If any arc on the $\overline{Z-Op}$ path reaches a bound at the same time or before an arc on the $\overline{Z-Dq}$ path, then the highest such arc on the $\overline{Z-Op}$ path is the outgoing arc.

The following lemma will be used to prove that the foregoing rules preserve the GAP basic structure.

Lemma 2: If Case I.B. or II.B. applies, then the pivot is always nondegenerate.

Proof: In Case I.B., an arc on the $\overline{Z-Dq}$ path provides the limitation on the flow change Δ . But since each 0-D link has lower leeway and is being decreased

by Δ on this path, and since each D-0 link has upper leeway and is being increased by Δ on this path, the value of Δ allowed by this path must be positive. Similar analysis applies to Case II.B., interchanging the roles of increases and decreases and applying them to the $\overline{Z-0p}$ path.

The significance of this lemma is apparent because if a shortcut procedure is constructed to identify cases I.B. and II.B., this provides a method of undertaking to make nondegenerate pivots.

Lemma 3: The rules I.A., I.B., II.A., II.B. preserve the GAP basis structure, and moreover, are the only rules that can if the root node is unchanged.

Proof: The lemma will be established for cases I.A. and I.B. The proof for cases II.A. and II.B. is similar. Suppose I.A. applies and the indicated rule is followed. It is important to note that maintaining the same root node implies that the only links which will change their O-D and D-0 link status in the new basic are the arcs in the $Z-0p$ path below the outgoing arc. Thus, the proof amounts to demonstrating that the links on the basic equivalent path satisfy the definition of a GAP basis after the basis exchange. First observe that all links above the discarded arc on the $\overline{Z-0p}$ path retain their orientation and their status, because the O-D links on this path only increase their flow and the D-0 links only decrease their flow, appropriately maintaining the lower and upper leeway conditions. Further, all arcs on the $\overline{Z-Dq}$ path, including (p,q) , retain their status by assumption since none were driven to a contrary bound. And since by assumption no arcs below the selected outgoing arc on the $Z-0p$ path are at their bounds, these arcs are double leeway arcs and thus qualify as arcs of the opposite status. Thus the standard rule of [10,13] for re-orienting the arcs of the $Z-0p$ path, and for attaching arc (p,q) as a D-0 link, preserves the GAP structure as desired.

On the other hand, if I.B. applies, then no arc on the $Z-Op$ path changes its status. Further, nothing above the outgoing arc on the $\overline{Z-Dq}$ path changes its status because no flows are driven to contrary bounds. But every arc below the outgoing arc can change its status because, by Lemma 2, the flow change is positive, and thus each of these arcs that previously had upper leeway now has lower leeway, and vice versa. Thus, once again, the rule of [10,13] that reverses the orientation of these arcs (if any exist--i.e., arc (p,q) itself or the first arc above it may be the outgoing arc) preserves the GAP structure. If arc (p,q) is not both the incoming and outgoing arc, then it receives the proper orientation as an O-D link. Finally, the foregoing observations disclose that no other choice of an outgoing arc will work, because it would create an O-D or D-0 link without the appropriate leeway on one of the two paths. This completes the proof.

5. GENERALIZED ALTERNATING PATH (GAP) ALGORITHM

On the basis of the preceding remarks, the rules of the GAP algorithm can be stated in an extremely simple fashion.

1. Select any feasible GAP basis for the capacitated transportation problem.
2. Successively apply the simplex pivot step keeping the root node fixed and picking the link to leave according to rules I and II.

The results previously established imply that the GAP algorithm will proceed through a sequence of GAP bases, bypassing all other basis structures. Further, these results show that a "next" GAP basis is always accessible to a given GAP basis, so that the method will not be compelled to stop prematurely without being able to carry out a pivot before the optimality criteria are satisfied.

The issue that remains to be resolved, then, is whether the method may progress through a closed circle of GAP bases without breaking out, and thus fail to converge. It will be shown that this cannot happen, and that, in fact, the GAP algorithm is finitely converging without any reliance upon external techniques such as perturbation, or lexicographical ordering, as is the ordinary simplex method. Further, this result does not require any restrictions on the choice of the incoming variable. Most importantly, it will be shown that *the form of convergence of the GAP algorithm has a particularly strong character, in which origin and destination node potentials each change in a uniform direction throughout any sequence of degenerate pivots.*

Our next result is the prelude to the main convergence theorem, which requires no use of perturbation and no restriction on the choice of the incoming arc.

Lemma 4: The pivot step that creates one GAP basis from another causes the node potentials to change in the following manner (holding the root node and its node potential constant):

(i) For cases I.A. and II.A.: The potentials for a subset of the origin nodes strictly decrease and the potentials for a subset of the destination nodes strictly increase, where at least one of these subsets is nonempty (the first for I.A. and the second for II.A.). All other node potentials are unchanged.

(ii) For cases I.B. and II.B.: The potentials for a subset of origin nodes strictly increase and the potentials for a subset of destination nodes strictly decrease. At least one of these subsets is nonempty (the first for II.B. and the second for I.B.), unless arc (p,q) is the outgoing arc, in which case they are both empty. All other potentials remain unchanged.

Proof: No node potentials change when arc (p,q) is both the incoming and outgoing arc. Thus assume that arc (p,q) is not the outgoing arc. In this case, as already discussed, the node potential values that change are restricted to those associated with subtree K . By this procedure, if subtree K contains the origin node of the entering arc then δ is subtracted from all origin node potentials in K and δ is added to all destination node potentials. On the other hand, if subtree K contains the destination node of the entering arc then δ is subtracted from all destination nodes and δ is added to all origin node potentials, where $\delta = \pi_{pq} = c_{pq} - R_p - K_q$. The assertions (i) and (ii) follow at once by observing that $0 \in K_p$ holds for cases I.A. and II.B. and $0 \in K_q$ in cases I.B. and II.A.

Our main result may be stated as follows:

Theorem: The primal GAP basis algorithm is finite and independent of the choice of incoming arc.

Proof: By Lemma 2, as long as case I.B. or II.B. holds, the method makes non-degenerate pivots, of which there are a finite number. We show that the number of degenerate pivots that can occur in unbroken succession is also finite. In particular, these pivots must all occur for case I.A. or II.A. But by Lemma 4, the node potentials are changing in a uniform direction throughout each step. Since the root node maintains a constant potential and the values of the other potentials are thus uniquely determined for any given basis, it follows that no basis can repeat during this succession of pivots, completing the proof.

Implementation Considerations

The rules I.A., I.B., II.A. and II.B. of the GAP transportation algorithm all identify the arc to leave the basis as either the highest or lowest quali-

ifying arc on a particular segment of the basis equivalent path, thus translating a complex set of interactions into a particularly simple prescription for implementation. This prescription is fully compatible with the use of the specialized labeling procedures developed for implementing primal network algorithms [4, 10, 15, 20] and consequently, the strong convergence property of the GAP algorithm is not gained at the expense of abandoning other means for accelerating the solution process.

Further, it is especially significant that in the only cases that afford the opportunity for a degenerate pivot (I.A. and II.A.), the arc to leave the basis is the lowest that qualifies on the specified path segment. This means that, by the customary trace of predecessors, the first arc encountered that determines the pivot to be degenerate is the one to drop. For practical problems, in which degenerate pivots have been reported to constitute 80% or more of the total iterations, this feature is particularly convenient. Moreover, there is no disadvantage to the rule that prescribes dropping the highest qualifying arc when the pivot is nondegenerate, because in this instance it is necessary to conduct a full trace of the arcs of the basis equivalent path in any event. Thus, in brief, the unique form of the GAP algorithm for the capacitated transportation problem is ideally suited to the use of implementation schemes designed to minimize the calculations at each iteration, while enjoying the benefits of the strong convergence property for circumscribing the total number of degenerate steps.

REFERENCES

1. Analysis, Research, and Computation, Inc., "Development and Computational Testing on Large Scale Primal Simplex Network Codes," ARC Technical Research Report, P.O. Box 4067, Austin, TX 78765 (1974).
2. R. S. Barr, "Streamlining Primal Simplex Transportation Codes," Research Report to appear, Center for Cybernetic Studies, University of Texas, Austin, Texas.
3. R. S. Barr, F. Glover, and D. Klingman, "An Improved Version of the Out-of-Kilter Method and a Comparative Study of Computer Codes," *Mathematical Programming*, 7, 1, 60-87 (1974).
4. R. S. Barr, F. Glover, and D. Klingman, "Enhancements to Spanning Tree Labeling Procedures for Network Optimization," Research Report CCS 262, Center for Cybernetic Studies, University of Texas at Austin, Austin, Texas 78712 (1976).
5. R. S. Barr, F. Glover, and D. Klingman, "The Alternating Basis Algorithm for Assignment Problems," Research Report CCS 263, Center for Cybernetic Studies, University of Texas at Austin, Texas 78712 (1976).
6. R. S. Barr, F. Glover, and D. Klingman, "A New Alternating Basis Algorithm for Semi-Assignment Networks," Research Report CCS 264, Center for Cybernetic Studies, University of Texas at Austin, Austin, Texas 78712 (1976).
7. G. E. Bennington, "An Efficient Minimal Cost Flow Algorithm," *Management Science*, 19, 9, 1021-1051 (1973).
8. G. Bradley, G. Brown, G. Graves, "Tailoring Primal Network Codes to Classes of Problems with Common Structure," ORSA/TIMS conference, Las Vegas (1975).
9. W. H. Cunningham, "A Network Simplex Method," Technical Report No. 207, Department of Mathematical Sciences, John Hopkins University (1974).
10. F. Glover, D. Karney, and D. Klingman, "Augmented Predecessor Index Method for Location Stepping-Stone Paths and Assigning Dual Prices in Distribution Problems," *Transportation Science*, 6, 1, 171-181 (1972).
11. F. Glover, D. Karney, and D. Klingman, "Implementation and Computational Study on Start Procedures and Basic Change Criteria for a Primal Network Code," *Networks*, 4, 3, 191-212 (1974).
12. F. Glover, D. Karney, D. Klingman, and A. Napier, "A Computational Study on Start Procedures, Basis Change Criteria, and Solution Algorithms for Transportation Problems," *Management Science*, 20, 6, 793-819 (1974).

13. F. Glover and D. Klingman, "Locating Stepping-Stone Paths in Distribution Problems Via the Predecessor Index Method," Transportation Science, 4, 220-26 (1970).
14. F. Glover and D. Klingman, "Omissions and Practical Misconceptions in the Hatch Article Comparing Network Algorithms," Research Report CCS 270, Center for Cybernetic Studies, University of Texas at Austin, Austin, Texas 78712 (1976).
15. F. Glover, D. Klingman, and J. Stutz, "Augmented Threaded Index Method for Network Optimization," INFOR, 12, 3, 293-298 (1974).
16. R. S. Hatch, "Bench Marks Comparing Transportation Codes based on Primal Simplex and Primal-Dual Algorithms," Operations Research, 23, 6, 1167-1171 (1975).
17. D. Klingman, A. Napier, and J. Stutz, "NETGEN-A Program for Generating Large Scale (Un)Capacitated Assignment, Transportation, and Minimum Cost Flow Network Problems," Management Science, 20, 5, 814-822 (1974).
18. J. Mulvey, "Column Weighting Factors and Other Enhancements to the Augmented Threaded Index Method for Network Optimization," Joint ORSA/TIMS Conference, San Juan, Puerto Rico (1974).
19. V. Srinivasan and G. L. Thompson, "Benefit-Cost Analysis of Coding Techniques for the Primal Transportation Algorithm," JACM, 20, 194-213 (1973).
20. V. Srinivasan and G. L. Thompson, "Accelerated Algorithms for Labeling and Relabeling of Trees with Application for Distribution Problems," JACM, 19, 4, 712-726 (1972).

Unclassified

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Center for Cybernetic Studies ✓ The University of Texas		2a. REPORT SECURITY CLASSIFICATION Unclassified	
		2b. GROUP	
3. REPORT TITLE The Generalized Alternating Path Algorithm for Transportation Problems			
4. DESCRIPTIVE NOTES (Type of report and, inclusive dates)			
5. AUTHOR(S) (Last name, middle initial, first name) Richard S. Barr Fred Glover Darwin Klingman			
6. REPORT DATE February 1977		7a. TOTAL NO. OF PAGES 19	7b. NO. OF REFS 20
8a. CONTRACT OR GRANT NO. N00014-76-C-0383; N00014-75-C-0616;0569		9a. ORIGINATOR'S REPORT NUMBER(S) Center for Cybernetic Studies Research Report CCS 282 ✓	
b. PROJECT NO. NR047-021		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
c.			
d.			
10. DISTRIBUTION STATEMENT This document has been approved for public release and sale; its distribution is unlimited.			
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY Office of Naval Research (Code 434) Washington, D.C.	
13. ABSTRACT A new primal extreme point algorithm for solving capacitated transportation problems is developed in this paper. This algorithm, called the generalized alternating path (GAP) algorithm, is a special purpose method specifically designed to take advantage of the bipartite structure and the often pervasive primal degeneracy of transportation problems.			

~~Classified~~
Security Classification

DD FORM 1473 (BACK)
1 NOV 65
S/N 0102-014-6800

Security Classification

A-31409